

PERANCANGAN APLIKASI PENGENALAN CITRA WAJAH MENGGUNAKAN METODE COMPLETE KERNEL FISHER DISCRIMINANT

Aninda Muliani¹, Suwandy Kosasih²

Program Studi Teknik Informatika, Universitas Prima Indonesia

Jl. Sekip Simpang Sikambing Medan

¹anindamh@gmail.com, ²wguo.wg@gmail.com

Abstrak - Pelacakan dan pengenalan wajah manusia (*face recognition*) merupakan salah satu bidang penelitian yang penting dan belakangan ini banyak aplikasi yang dapat menerapkannya, baik di bidang komersial maupun bidang penegakan hukum. Teknik pengenalan wajah pada saat ini telah mengalami kemajuan yang sangat berarti, mengingat teknik pengenalan wajah ini merupakan bidang penelitian yang sangat dibutuhkan untuk berbagai bidang. Aplikasi *face recognition* pada saat ini banyak dikembangkan karena dapat diaplikasikan di berbagai bidang permasalahan seperti pengenalan kriminal, aplikasi keamanan, absensi, ataupun interaksi manusia dengan komputer. Banyak metode yang digunakan dalam aplikasi pengenalan wajah, salah satu diantaranya adalah metode *Complete Kernel Fisher Discriminant* (CKFD) yang merupakan pengembangan dari metode *Kernel Principal Component Analysis* (PCA) dan *Fisher Discriminant Analysis* (FDA). Metode *Complete Kernel Fisher Discriminant* (CKFD) memiliki dua keuntungan bila dibandingkan dengan *Kernel Fisher Discriminant* (KFD) sebelumnya. Pertama, implementasi algoritma ini dapat dibagi ke dalam dua fase, yaitu *Kernel principal component analysis* (KPCA) ditambah *Fisher linear discriminant analysis* (FLD) sehingga hasilnya lebih transparan dan lebih sederhana. Kedua, CKFD bisa membuat dua kategori informasi diskriminan sehingga hasilnya lebih kuat. Perancangan aplikasi ini bertujuan untuk implementasi sistem pengenalan wajah menggunakan metode *Complete Kernel Fisher Discriminant* (CKFD). Perancangan aplikasi pengenalan citra wajah ini dirancang menggunakan Visual Basic 2008, dimana sistem ini sudah bisa digunakan untuk pengenalan wajah dengan rata-rata 56,48% sehingga hasilnya dapat dikatakan tidak begitu akurat. Namun sistem ini perlu pengembangan lebih lanjut, agar bisa digunakan untuk mengenali citra wajah dengan lebih akurat, serta peningkatan ukuran dan resolusi gambar yang digunakan di atas 240x240 piksel.

Kata Kunci - Sistem Pengenalan Wajah dan Metode Complete Kernel Fisher Discriminant (CKFD)

Abstract - Face recognition is one of the important research fields and lately many applications can apply it, both in the commercial and law enforcement fields. Facial recognition techniques have now made significant progress, considering that face recognition techniques are a much-needed field of research for various fields. The face recognition application is currently being developed because it can be applied in various fields of problems such as criminal recognition, security applications, attendance, or human interaction with computers. Many methods are used in face recognition applications, one of which is the *Complete Kernel Fisher Discriminant* (CKFD) method which is the development of the *Kernel Principal Component Analysis* (PCA) method and *Fisher Discriminant Analysis* (FDA). The *Complete Kernel Fisher Discriminant* (CKFD) method has two advantages compared to the previous *Kernel Fisher Discriminant* (KFD). First, the implementation of this algorithm can be divided into two phases, namely *Kernel principal component analysis* (KPCA) plus *Fisher linear discriminant analysis* (FLD) so that the results are more transparent and simpler. Second, the CKFD can make two categories of discriminant information so that the results are stronger. The design of this application aims to implement face recognition systems using the *Complete Kernel Fisher Discriminant* (CKFD) method. The design of face image recognition application is designed using Visual Basic 2008, where this system can be used for face recognition with an average of 56.48% so the results can be said to be not very accurate. But this system needs further development, so that it can be used to recognize facial images more accurately, and increase the size and resolution of images used above 240x240 pixels.

Keywords - Face Recognition System, and Method Complete Kernel Fisher Discriminant (CKFD)

I. PENDAHULUAN

A. Latar Belakang

Pelacakan dan pengenalan wajah manusia (*face recognition*) merupakan salah satu bidang penelitian yang penting, dan belakangan ini banyak aplikasi yang dapat menerapkannya, baik di bidang komersial maupun bidang penegakan hukum. Teknik pengenalan wajah pada saat ini telah mengalami kemajuan yang sangat berarti,

mengingat teknik pengenalan wajah ini merupakan bidang penelitian yang sangat dibutuhkan untuk berbagai bidang. Aplikasi *face recognition* pada saat ini banyak dikembangkan karena dapat diaplikasikan di berbagai bidang permasalahan seperti pengenalan kriminal, aplikasi keamanan, absensi, ataupun interaksi manusia dengan komputer. Namun untuk mengembangkan model komputasi pengenalan wajah ini cukup sulit, karena wajah manusia mempresentasikan sesuatu

yang bersifat kompleks, sehingga untuk mengembangkan model komputasi yang ideal untuk pengenalan wajah manusia adalah sesuatu hal yang sulit. Dan ada juga masalah lain yang dihadapi oleh aplikasi pengenalan wajah, dikarenakan wajah manusia yang memiliki ekspresi beragam, perubahan atribut seperti kacamata, kumis, janggut dan juga kualitas gambar yang ditangkap, kecerahan dan pencahayaan gambar yang kurang baik juga dapat mempengaruhi aplikasi dalam pengenalan wajah.

Banyak metode yang digunakan dalam aplikasi pengenalan wajah, salah satu diantaranya adalah metode *Complete Kernel Fisher Discriminant* (CKFD) yang merupakan pengembangan dari metode *Kernel Principal Component Analysis* (PCA) dan *Fisher Discriminant Analysis* (FDA). Metode *Complete Kernel Fisher Discriminant* (CKFD) memiliki dua keuntungan bila dibandingkan dengan *Kernel Fisher Discriminant* (KFD) sebelumnya. Pertama, implementasi algoritma ini dapat dibagi ke dalam dua fase, yaitu *Kernel principal component analysis* (KPCA) di tambah *Fisher linear discriminant analysis* (FLD) sehingga hasilnya lebih transparan dan lebih sederhana. Kedua, CKFD bisa membuat dua kategori informasi diskriminan sehingga hasilnya lebih kuat.

Untuk mengatasi permasalahan pendeteksian dan pengenalan wajah seperti penjabaran di atas, maka penulis berkeinginan untuk merancang sebuah aplikasi pengenalan wajah yang dapat digunakan untuk mengenali wajah dengan membandingkannya dengan data yang tersimpan dalam komputer dengan mengangkat judul “*Perancangan Aplikasi Pengenalan Citra Wajah Menggunakan Metode Complete Kernel Fisher Discriminant (CKFD)*” dalam penulisan skripsi ini.

B. Rumusan Masalah

Berdasarkan latar belakang pemilihan judul, maka yang menjadi permasalahan adalah:

1. Bagaimana menerapkan metode *Complete Kernel Fisher Discriminant* untuk mengenali citra wajah ?
2. Bagaimana melakukan pengujian terhadap metode *Complete Kernel Fisher Discriminant* ?

C. Tujuan dan Manfaat Penulisan

Tujuan penelitian ini adalah:

1. Untuk memahami proses penerapan metode *Complete Kernel Fisher Discriminant* (CKFD) dalam aplikasi pengenalan wajah.
 2. Untuk merancang aplikasi pengenalan wajah dengan metode *Complete Kernel Fisher Discriminant* (CKFD).
 3. Aplikasi pengenalan wajah dapat digunakan sebagai referensi dalam pengembangan aplikasi pengenalan wajah yang lebih kompleks.
- Manfaat penelitian ini adalah sebagai berikut :
1. Membantu pemahaman metode *Complete Kernel Fisher Discriminant* (CKFD).
 2. Menambah referensi bagi mahasiswa lainnya yang ingin mempelajari mengenai pengenalan pola yang mendukung pembelajaran mata kuliah kecerdasan buatan, pengolahan citra, serta teknologi *biometric* wajah.
 3. Menjadi bahan referensi untuk melakukan penelitian lebih lanjut tentang aplikasi pengenalan wajah.
 4. Mengetahui proses pencocokkan wajah dalam aplikasi pengenalan wajah.

D. Batasan Masalah

Batasan masalah dalam pembahasan penelitian ini adalah sebagai berikut :

1. Metode yang digunakan dalam pengenalan wajah adalah metode *Complete Kernel Fisher Discriminant* (CKFD).
2. Pengenalan wajah akan menggunakan bantuan perhitungan *Threshold*.

II. KAJIAN PUSTAKA

A. Metode Kernel PCA

Representasi linear data pada PCA seringkali mengalami kesulitan untuk memodelkan data yang sangat kompleks. Kernel PCA (KPCA) merupakan pengembangan non-linear dari PCA, sehingga data lebih mudah dimodelkan dalam representasinya. Metode KPCA menghitung PCA di feature space, dimana data x dipetakan melalui sebuah fungsi non-linear $\phi(x)$ dan dekomposisi eigenvalue dilakukan terhadap matriks kovarian dari data

yang telah dipetakan sesuai dengan persamaan berikut.

$$C^\phi = \frac{1}{n} \sum_{i=1}^n \phi(X_i) \phi(X_i)^T \quad \dots\dots(1)$$

Fungsi pemetaan non-linier mengakibatkan penambahan jumlah dimensi yang cukup besar dan kadangkala menjadi tak terhingga pada data yang telah dipetakan. Perkalian titik pada feature space ini dapat digantikan dengan fungsi kernel

$k(x_i, x_i) = \phi(x_i) \phi(x_i)^T$ yang dapat dievaluasi pada input space. Cara ini disebut sebagai kernel trick yang dapat dipakai untuk semua algoritma dengan memformulasikan perkalian titik feature space. Fungsi kernel ini, digunakan untuk membentuk matriks kernel.

$$K = \begin{bmatrix} k(x_1, x_1) & \dots & k(x_1, x_n) \\ \vdots & \ddots & \vdots \\ k(x_n, x_1) & \dots & k(x_n, x_n) \end{bmatrix} \quad \dots\dots\dots(2)$$

Persamaan 2 nilai K merupakan pengganti matriks kovarian pada PCA. Dekomposisi eigenvalue akan dilakukan terhadap matriks kernel tersebut, sehingga menghasilkan fungsi.

$$\alpha_1 = \left[\alpha_1^{(k)}, \dots, \alpha_n^{(k)} \right]^T \quad (k=1, \dots, n) \quad \dots\dots\dots(3)$$

Persamaan 3 nilai merupakan nilai eigenvector dari nilai eigenvalue hasil dekomposisi, sehingga representasi data baru untuk setiap citra dapat diperoleh menggunakan persamaan tersebut.

B. Complete Kernel Fisher Discriminant (CKFD)

Metode Complete Kernel Fisher Discriminant Analysis (CKFD) mengaplikasikan kernel trick non linier sebelum menerapkan aturan diskriminan FLDA. Metode CKFD, awalnya memang ditujukan untuk mengatasi kompleksitas yang terjadi pada citra wajah, sehingga masih menyisakan problema singularitas (*small training sample problem*) (Rahayu, 2015 : i).

Usaha untuk memperbaiki performansi CKFDA, bisa dilakukan dengan berbagai cara, yaitu dengan inovasi kernel, aturan diskriminan baru, penggunaan classifier berbeda maupun kombinasi diantaranya. Dalam penelitian ini dikaji

penggunaan kernel *spectrally weighted* (SW) yang diturunkan dari kernel 94ctor94mial dan radial based function (RBF), dalam mekanisme SVM classifier. Dengan menerapkan algoritma dekomposisi LibSVM berbasis gradient descent, diharapkan bisa mendapatkan optimasi bobot yang bisa meningkatkan performansi laju pengenalan (*recognition rate*).

Metode *Complete Kernel Fisher Discriminant* (CKFD) memiliki dua keuntungan bila dibandingkan dengan *Kernel Fisher Discriminant* (KFD) sebelumnya. Pertama, implementasi algoritma ini dapat dibagi ke dalam dua fase, yaitu *Kernel principal component analysis* (KPCA) di tambah *Fisher linear discriminant analysis* (FLD) sehingga hasilnya lebih transparan dan lebih sederhana. Kedua, CKFD bisa membuat dua kategori informasi diskriminan sehingga hasilnya lebih kuat.

Di dalam transformasi ruang KPCA, maka dapat dibagi ke dalam dua kelas yaitu between class dan within class scatter matriks (S_b, S_w). Adapun rumus yang digunakan adalah sebagai berikut :

$$S_b = \sum_{i=1}^e P(\omega_i) (m_i - m_o) (m_i - m_o)^T \quad \dots\dots\dots(4)$$

$$S_w = \sum_{i=1}^e P(\omega_i) \left\{ \sum_{j=1}^{l_i} \frac{1}{l_i} (X_{ij} - m_i) (X_{ij} - m_i)^T \right\} \quad \dots\dots(5)$$

Di mana :

X_{ij} : Menandakan j sebagai sampel *training* dari *class* i.

$P(\omega_i)$: Probabilitas *class* i sebelumnya.

l_i : Jumlah sampel *training* dalam *class* i.

m_i : Rata-rata vector sampel *training* dalam *class* i.

m_o : Rata-rata vector the mean vector semua sampel *training*.

Algoritma CKFD secara lengkap adalah sebagai berikut :

1. Gunakan KPCA untuk transformasi data *space* asli (X-Space) ke dalam sebuah m-dimensial space R^m (Y-Space), di mana $m=M-1$, di mana M adalah jumlah sampel *training*.
2. Dalam R^m , Tentukan between class dan within class scatter matriks S_b dan S_w . Hitung orthonormal eigenvector S_w berupa y_1, \dots, y_m dengan asumsi, pertama mempunyai nilai eigen positif.
3. Ekstrak fitur diskriminan I : di mana $P_I = (y_1, \dots, y_q)$. Tentukan

$\bar{S}_b = P_1^T S_b P_1$, $\bar{S}_w = P_1^T S_w P_1$ dan hitung l sebagai eigenvector umum u_1, \dots, u_l dari $S_b \bar{S}_w^{-1}$ menggunakan algoritma :

KPCA dalam input space $R^n \Leftrightarrow$ PCA dalam feature space R^N

Fitur diskriminan kategori I adalah

$$Z_j = (P_2 v_j)^T Y, j = 1, \dots, l,$$

di mana $l = c$, c adalah jumlah *class*.

4. Ekstrak fitur diskriminan kategori II: Di mana $P_2 = (y_{q+1}, \dots, y_m)$. Tentukan $\bar{S}_b = P_2^T S_b P_2$, dan hitung \bar{S}_b pertama $d-1$ sebagai eigenvalue terbesar dan sesuaikan dengan eigenvector v_{l+1}, \dots, v_d . Fitur diskriminan kategori II adalah $Z_j = (P_2 v_j)^T Y, j = l+1, \dots, d$, di mana d adalah jumlah fitur yang diinginkan (Yang, dkk, 2003 : 4).

Dari Algoritma CKFD, dua kategori fitur diskriminan dihasilkan. Total jumlah fitur diskriminan bisa di atas $c-1$ dan hingga $2(c-1)$, di mana c adalah jumlah *class*. Ini berbeda dari algoritma KFD yang sudah ada, yang dapat menghasilkan fitur diskriminan paling banyak $c-1$. Pada dasarnya, analisis diskriminan umum (GDA) adalah setara dengan CKFD hanya menggunakan fitur diskriminan dari Kategori I (dari Langkah 1 ke Langkah 3). Tapi, GDA tidak transparan dan sederhana seperti CKFD (Yang, dkk, 2003 : 4-5).

Tahapan proses pengenalan wajah dapat dirincikan sebagai berikut:

1. Preprocessing

Citra digital yang di muat memiliki intensitas *pixel* yang beragam. Tahap *preprocessing* berguna untuk membatasi intensitas *pixel* citra. Sedangkan tahap-tahap yang dilakukan dalam *preprocessing* adalah sebagai berikut :

1. Proses konversi citra *input* ke bentuk *grayscale*

Sebelum citra di proses dengan masing-masing algoritma, citra wajah di proses dengan proses *training*. Proses *training* dilakukan dengan mengubah *image* RGB menjadi *image grayscale* untuk mendapatkan sebuah vektor ciri dan mendapatkan jarak *distance* yang nantinya akan

digunakan pada proses pengenalan dengan masing-masing algoritma.

Tahap *grayscale* citra adalah kegiatan untuk menyederhanakan model citra. Tiga *layer* pada citra berwarna, *R-layer*, *G-layer*, dan *B-layer* di ubah menjadi satu *layer* grayscale. Untuk mengubah citra berwarna yang memiliki nilai matrik masing-masing R, G, dan B menjadi citra grayscale dengan membagi jumlah ketiga *layer*, dengan persamaan di bawah ini :

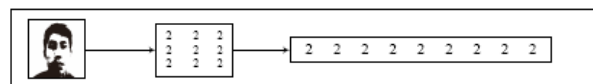
$$f_0(x, y) = \frac{f_i^R(x, y) + f_i^G(x, y) + f_i^B(x, y)}{3}$$

2. Proses Thresholding

Proses ini dilakukan untuk mengubah citra berderajat keabuan menjadi citra biner atau hitam putih. Intensitas *pixel* citra hasil *grayscale* dibandingkan dengan nilai ambang (*threshold*). Jika nilai *pixel* lebih besar dari nilai ambang, maka *pixel* akan direpresentasikan dengan warna putih. Jika nilai *pixel* lebih kecil daripada nilai ambang, maka *pixel* akan direpresentasikan dengan warna hitam.

3. Proses Reshape Image 2D ke Vektor ID

Setelah didapatkan *image* yang hasil *threshold* selanjutnya ubah menjadi vektor satu dimensi untuk untuk mempermudah dalam menghitung *distance* dari setiap *image* yang nantinya digunakan dalam proses pengenalan. Misalnya dalam *training image* terdapat *image* dengan ukuran 3 x 4 piksel maka kita akan mempunyai *flatvector* ukuran 1 x 9. Ilustrasinya sebagai berikut :



Gambar 3.3 Proses *Reshape* 2D ke 1D Vektor *Image*

2. Algoritma CKFD

Dari Algoritma CKFD, dua kategori fitur diskriminan dihasilkan. Total jumlah fitur diskriminan bisa di atas $c-1$ dan hingga $2(c-1)$, di mana c adalah jumlah *class*. Ini berbeda dari algoritma KFD yang sudah

ada, yang dapat menghasilkan fitur diskriminan paling banyak c-1.

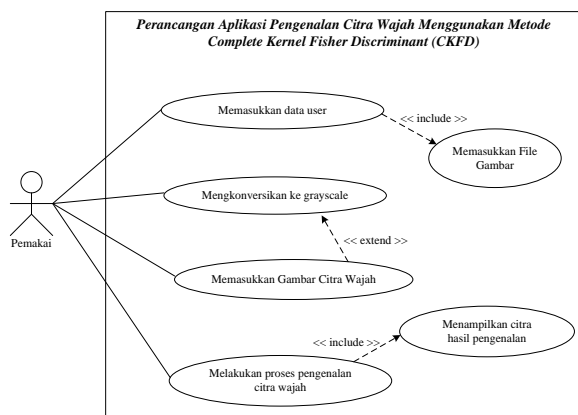
3. Identifikasi

Proses terakhir adalah identifikasi yaitu dengan memproyeksikan *test image* ke *face space* dan menghitung *score*. Proses identifikasi dapat dirincikan sebagai berikut:

1. Load semua wajah yang sudah diproyeksikan ke *database*.
2. Lakukan operasi pengurangan, proj dengan semua wajah yang telah diproyeksikan. Ambil nilai absolutnya dan jumlahkan, hasil adalah *distance* menggunakan jarak *euclidean*.
3. Hitung jarak *distance* antara bobot *image test* dan bobot *training set*.
4. Ambil *distance* terkecil sebagai hasil dari wajah yang telah diproyeksikan. Wajah ini menjadi hasil identifikasi.

3. METODE

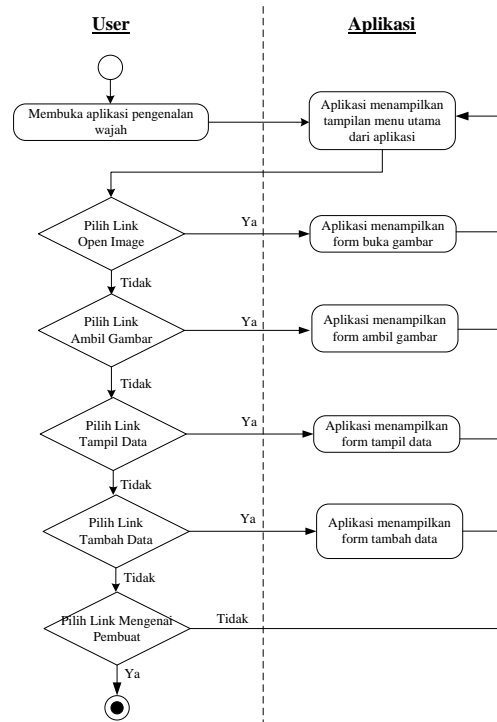
Untuk menjelaskan bagaimana bentuk interaksi *input* dan *output* pada perangkat lunak yang dirancang, penulis merancang sebuah *use case diagram* sebagai alat bantu, sebagaimana terlihat pada Gambar 1.



Gambar 1. Use Case Diagram dari Aplikasi Pengenalan Citra Wajah

Pada Gambar 1 di atas, *use case diagram* interaksi aplikasi pada perangkat lunak pengenalan wajah dengan menggunakan metode CKFD ini memiliki empat proses utama, yaitu memasukkan file gambar, memasukkan perintah ekstrak, memasukkan perintah simpan dan menampilkan hasil.

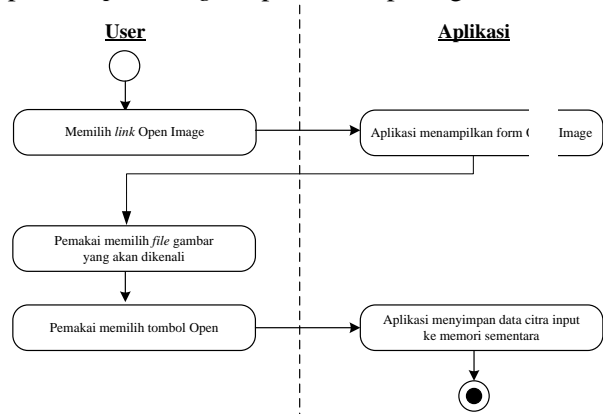
Activity Diagram dari aplikasi pembelajaran dapat dilihat pada gambar 2 dibawah.



Gambar 2. Rancangan Activity Diagram Menu dari Aplikasi Pengenalan Citra Wajah

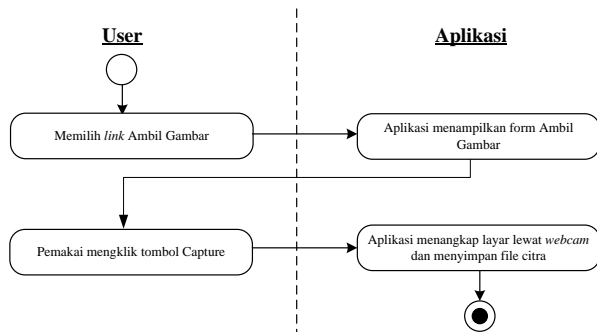
Pada saat pemakai membuka aplikasi, maka perangkat lunak akan menampilkan *form* utama yang berisi *link* yang dapat dipilih yaitu *link open image*, *link* ambil gambar, *link* tampil data, *link* tambah data, *link* mengenai pembuat.

Activity diagram yang menggambarkan proses *open image* dapat dilihat pada gambar 3



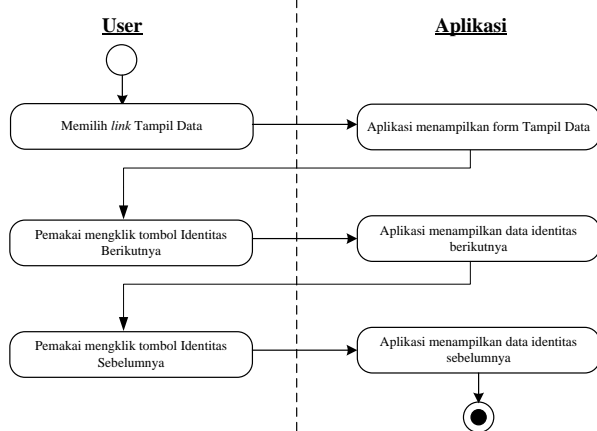
Gambar 3. Rancangan Activity Diagram Menu dari Proses Open Image

Activity diagram yang menggambarkan proses ambil gambar dapat dilihat pada gambar 4.



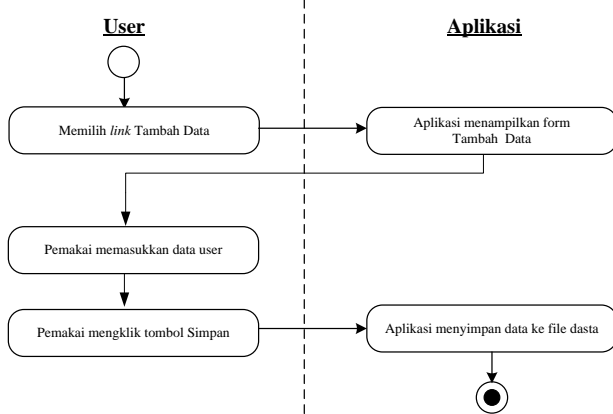
Gambar 4. Rancangan Activity Diagram Menu dari Ambil Gambar

Activity diagram yang menggambarkan proses tampil data dapat dilihat pada gambar 5.



Gambar 5. Rancangan Activity Diagram Menu dari Tampil Data

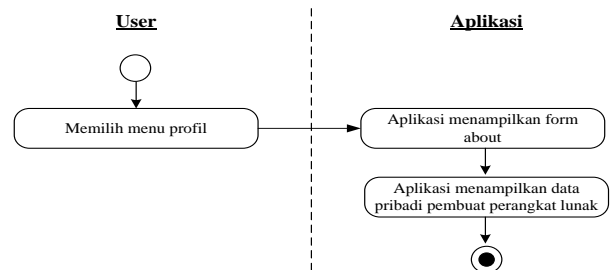
Activity diagram yang menggambarkan proses tambah data dapat dilihat pada gambar 6.



Gambar 6. Rancangan Activity Diagram Menu dari Tambah Data

Rancangan activity diagram yang menggambarkan proses mengenai pembuat dapat dilihat pada gambar 7. Setelah user memilih form

mengenai pembuat maka sistem akan menampilkan informasi mengenai pembuat aplikasi simulasi.



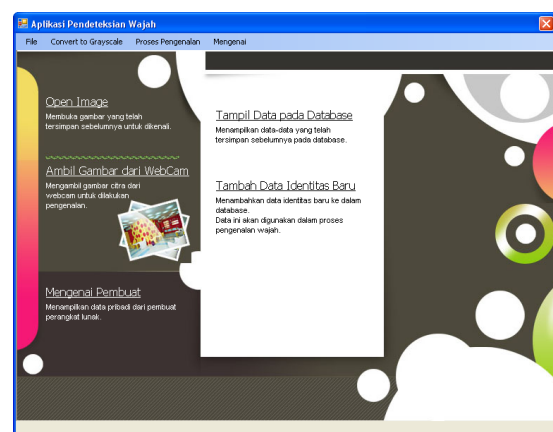
Gambar 7. Rancangan Activity Diagram untuk Proses Tentang Pembuat

4. HASIL DAN PEMBAHASAN

Berdasarkan hasil implementasi sistem pengenalan wajah menggunakan metode CKFD ini maka tampilan program yang digunakan adalah sebagai berikut :

1. Tampilan Menu Utama

Form menu utama merupakan form induk tempat pengguna berinteraksi dengan program yang dibuat. Form menu utama terdiri dari menu Fitur, menu Pengenalan dan Keluar. Menu Fitur digunakan untuk menampilkan Form Fitur, menu Pengenalan untuk menampilkan form Kenal, dan menu Keluar untuk keluar dari sistem yang dibuat. Tampilan Form Menu Utama dapat dilihat pada Gambar 8.



Gambar 8. Tampilan Form Utama

Pada tampilan utama ini terdapat beberapa menu yang berfungsi untuk mengakses form-form yang terdapat dalam sistem. Berikut perincian dari menu yang terdapat dalam sistem:

1. Menu File, terdiri dari:

1. Submenu New, yang berfungsi untuk mengosongkan citra yang ditampilkan.
2. Submenu Open Image, yang berfungsi untuk menampilkan kotak dialog open untuk memilih file citra yang ingin dicocokkan.
3. Submenu Save Image, yang berfungsi untuk menyimpan citra yang sedang ditampilkan.
4. Submenu Exit, yang berfungsi untuk menutup aplikasi.

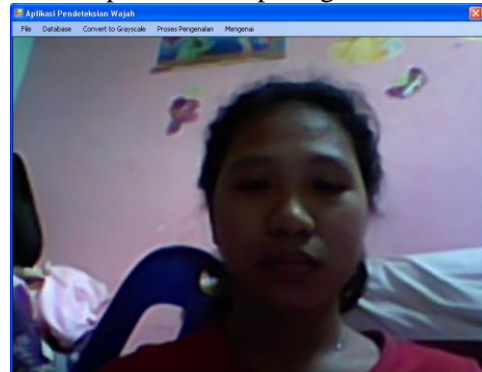
2. Tampilan Form Tambah Identitas
Tampilan Form Tambah Identitas digunakan untuk menambahkan data identitas baru ke dalam *file* data. Form fitur ini digunakan melakukan penyimpanan file citra wajah yang digunakan sebagai fitur pengenalan wajah menggunakan metode CKFD. Tampilan Form Tambah Identitiats dapat dilihat pada Gambar 9.

Gambar 9. Tampilan Form Tambah Identitas

3. Tampilan Form 'Tampil'
Form ini berfungsi untuk menampilkan semua daftar identitas yang terdaftar pada *database*. Tampilan *form* Tampil dapat dilihat pada gambar berikut:

Gambar 10. Tampilan *Form* Tampil

4. Tampilan Form Pengenalan
Pilihlah gambar yang ingin dicocokkan. Setelah itu, klik tombol 'Open'. Sistem akan menampilkan gambar yang akan dicocokkan pada form Main seperti terlihat pada gambar berikut:



Gambar 11. Form Main Setelah Pemilihan Gambar Pengujian

Setelah mengkonversi ke bentuk *grayscale*, maka dapat dilakukan melakukan proses pengenalan. Kliklah menu Proses Pengenalan sehingga sistem akan menampilkan hasil pengenalan seperti terlihat pada gambar berikut:

Gambar 12. Form Hasil Pengenalan untuk Pengujian Berhasil

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil penelitian yang dilakukan, maka penulis menarik kesimpulan sebagai berikut :

1. Jumlah sampel yang digunakan harus banyak. Hal ini diperlukan agar dapat meningkatkan akurasi hasil pengenalan. Namun, jumlah sampel yang banyak

- akan mengakibatkan proses pengenalan wajah menjadi lama.
2. Metode yang digunakan tidak dapat mengenali gambar wajah yang dimasukkan apabila ukuran kepala pada gambar tersebut berbeda dengan ukuran kepala pada gambar di *database*.

B. Saran

Sesuai dengan kesimpulan di atas, maka penulis memberikan saran sebagai berikut :

1. Untuk meningkatkan akurasi dari metode yang digunakan, maka dapat dikombinasikan dengan algoritma pendeteksian wajah sehingga proses pengenalan wajah hanya terhadap daerah wajah saja.
2. Mengoptimalkan waktu eksekusi dengan memperbaiki algoritma atau dengan menggunakan bahasa pemrograman lainnya.
3. Sistem perlu dilakukan perbandingan dengan metode pengenalan wajah lain, seperti Template Matching, Feature Extraction, dan lain sebagainya sehingga dapat diketahui peroleh metode terbaik digunakan untuk pengenalan wajah.

DAFTAR PUSTAKA

- [1] Darmayudha, I Ketut, 2011, Pemrograman Aplikasi Database dengan Microsoft Visual Basic.Net 2008, Penerbit Informatika, Bandung.
- [2] Kadir, A. dan Susanto, 2013, Teori dan Aplikasi Pengolahan Citra, Penerbit Andi, Yogyakarta.
- [3] Kurniawan, Dwi Ely, 2012, Rancang Bangun Sistem Pengenalan Wajah Menggunakan Filter Gabor, Program Studi Magister Sistem Informasi, Universitas Diponegoro, Semarang.
- [4] Ladjamuddin, Al Bahra Bin, 2011, Rekayasa Perangkat Lunak, Penerbit Graha Ilmu, Yogyakarta.
- [5] Rahayu, Sri, 2005, Penerapan Fungsi Adaptif Pada Kernel Fisher Discriminant Analysis (KFDA) Untuk Pengenalan Wajah, Program Studi Master, Institut Teknologi Surabaya (ITS), Surabaya.
- [6] Yang, dkk, 2005, KPCA Plus LDA: A Complete KernelFisher Discriminant Framework forFeature Extraction and

Recognition, Actions On Pattern Analysis And Machine Intelligence, Vol. 27, No. 2, Published by the IEEE Computer Society.